

# **A Novel Method for Reconstructing Broken Contour Lines Extracted from Scanned Topographic Maps**

**Wang Feng,<sup>1</sup> Liu Pingzhi,<sup>2</sup> Yang Yun,<sup>2</sup> Wei Haiping<sup>3</sup> and AN Xiaoya<sup>2</sup>**

1. Xi'an Research Institute of Surveying and Mapping, Xi'an, Shan'xi, China, China; bjzxgsb@163.com

2. Xi'an Research Institute of Surveying and Mapping, Xi'an, Shan'xi, China; liupingzhi2017 @163.com, yytoall @126.com, axy2001 @tom.com

3. Zhengzhou Institute of Surveying and Mapping, Xi'an, Shan'xi, China; haipingwei @163.com

**Abstract:** It is known that after segmentation and morphological operations on scanned topographic maps, gaps occur in contour lines. It is also well known that filling these gaps and reconstruction of contour lines with high accuracy and completeness is not an easy problem. In this paper, a novel method is proposed dedicated in automatic or semiautomatic filling up caps and reconstructing broken contour lines in binary images. The key part of end points' auto-matching and reconnecting is deeply discussed after introducing the procedure of reconstruction, in which some key algorithms and mechanisms are presented and realized, including multiple incremental backing trace to get weighted average direction angle of end points, the max constraint angle control mechanism based on the multiple gradient ranks, combination of weighted Euclidean distance and deviation angle to determine the optimum matching end point, bidirectional parabola control, etc. Lastly, experimental comparisons based on typically samples are complemented between proposed method and the other representative method, the results indicate that the former holds higher accuracy and completeness, better stability and applicability.

**Keywords:** scanned topographic maps, broken contour lines, reconstruction, points' auto-matching

## **1. Introduction**

Although photogrammetric interpretations of aerial photography and satellite imagery are the most common methods for extracting altimetry information, a great number of existing paper topographic maps, which remain the most frequent form of ground surface representation, provide a less expensive alternative: extracting contour lines from scanned topographic maps. One kind of the most interesting information shown by these paper-based topographic maps are the contour lines, which are imaginary lines that join different points located at the same elevation.

The aliasing and false colors are inevitable in scanned common-conditioned topographic map, a large number of intersects and overlaps among point, line and region features, contour lines are usually the lowest layer in topographic map, all of those problems make the segmentation and vectorization of scanned topographic maps even more difficult, and also result in a great amount of caps in contour lines' images, which could not meet the demands of automatic vectorization.

At present, existing techniques dedicated to realize the reconstruction of broken contour lines still could not give satisfactory results yet, and the primary defects lies in the low accuracy of end points' matching and reconnecting, lacking of stability and applicability for all kinds of scanned topographic maps(good, poor and worse). So the research on the reconstruction of broken contour lines is an open and interesting problem and field, it is a meaningful and significant work to bring forward an effective method to reconstruct broken contour lines, which could solve the problem of filling up the caps of broken contour lines exactly and completely.

In this paper, we first summarize the related works in section 2. Next, in section 3, introduce the general procedure of reconstructing broken contour lines, then design a novel method to match end points accurately and automat-

ically, a series of key algorithms and rules for supporting this method are proposed and realized. In section 4, full detailed processing flow is presented. In last section, there are some comparative experiments and analysis.

## 2. Related works

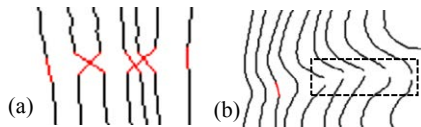
In general, vectorization of contour lines from scanned map needs four major steps: ①color segmentation for contour lines' image; ②binary, thinning and pruning shears; ③filling up caps in broken contour lines and reconstructing them; ④raster to vector conversion of all thinned and reconnected contour lines.

Despite most of the attentions have been put on step1 and step2 for a long time, a slightly greater attention has been paid to step3: automatic or semi-automatic reconstruction of broken contour lines. A great amount of professors have carried out a lot of research works about reconstruction of broken contour lines, and there are many papers have been published. This paper analyzed the latest related research.

When binary image is segmented from scanned map, Morphological filters is applied to the resulted binary image for noise reduction, and a reconnection algorithm based on local information is designed: firstly Distance is computed by combining Euclidean distances between extremities and differences between their directions, then extremities in lower distance are connected (Arrighi and Soille 1999). Another approach named A\* search algorithm is similar to Arrighi's method(Khotanzad and Zink 2003; Mun San et al. 2004). The images processed by those approaches still cause some problems such as overlapping contour lines and false connections (Figure 1(a)).

A line tracing technique for reconnecting the contour lines, in which an interruption is resolved by searching from an end point of a line and the corresponding end point might be found within a sector around the current direction(Eikvrl et al. 1995).

A semi-automatic method for broken contour lines' reconstruction based on orientation, angle and Euclidean distance is proposed (Ismail and Sheiriff 2006). But this method is not optimal and cannot reconnect some contours (Figure 1(b)).



**Fig. 1.** (a) Samples of intersecting connections of contour lines(Source: Khotanzad and Zink 2003). (b) Samples of gaps unsolved (Source: Ismail and Sheiriff 2006)

An approach based on Delaunay Triangulation and Voronoi Diagrams is proposed which uses the concept of medial axis of a curve  $\lambda$ : a point in the plane belongs to the medial axis of a curve  $\lambda$ , if at least two points located on the curve  $\lambda$  are at the same distance from this point(Amenta et al.1998; Ghircoias and Brad 2001). Its disadvantage is insufficient information and unsolved gaps.

An improved method on the basis of min pair of break-points algorithm is: colorate contour lines, build spatial grid-index, find connect point between two break points on the basis of different color and distance, then connect the connect-points(Huang et al. 2006).

GGVF (Generalized gradient vector flow) snake model algorithm is proposed to extract contour lines and use the directional field information to reconnect the broken contour lines(Xin et al. 2006). An automatic approach for filling up gaps between contour lines is: gradient orientation field generation from contour line normal; matching of end-points which is based on perfect matching algorithm; and filling the gap between each pair of end points(Pouderoux and Spinello 2007). The disadvantage of this approach is its processing time is very long for users.

An advanced semi-automatic approach is based on local and geometric properties, such as parabolic and opposite directions, the y-ordinate's differences of end points, back-tracing(Samet and Hancer 2012). But false connections and unsolved caps could not be avoided(Figure 2).



**Fig .2.** Samples of wrong connections and not filling caps (Source: Samet 2012)

Knowledge based on reconnection techniques was summarized, which aim at developing various solutions for re-connecting broken contour lines and compare their performance based on their ability to determine the search space for locating the extension point and its ability to reconnect(Pradhan et al. 2013)

A great amount of original scanned maps with high curve density are in poor or even worse quality. Reconnection errors (intersecting and overlapping) and unsolved gaps are still not handled with effectively and completely by using methods discussed above. In this paper, we proposed a novel reconstruction method based on a series of key algorithms and rules, which aims to be a semiautomatic procedure solution for reconnecting broken contour lines from binary images of scanned topographic maps, avoiding wrong reconnections and increasing completeness rate so as to reduce the human intervention as much as possible.

### 3 Proposed method

In this section, only the task of filling up the gaps and reconstruction of broken contour lines is handled with. It is assumed that other necessary tasks (topographic map digitization; color image segmentation and thresholding; filtering noisy pixels; thinning and pruning the binary image) are done before the task of filling up the gaps and reconnection of contour lines. In other words, an input for proposed method is post-segmentation, binary and thinned contour line image.

The reconstruction of broken contour lines includes three stages: ①end points detection; ②end points matching; ③reconnection of matched end points. The end points detection mean that the ending pixel has only one neighbor. End points matching mean that detecting and identifying the most suitable chosen end point to an unfilled end point. Reconnection of matched end points could be finish with Cubic Spline Interpolation or Newton Interpolation method(Parker 1997; Giesen 2000). In this paper, Newton Interpolation method which has effective performance and provides users smooth curves is used.

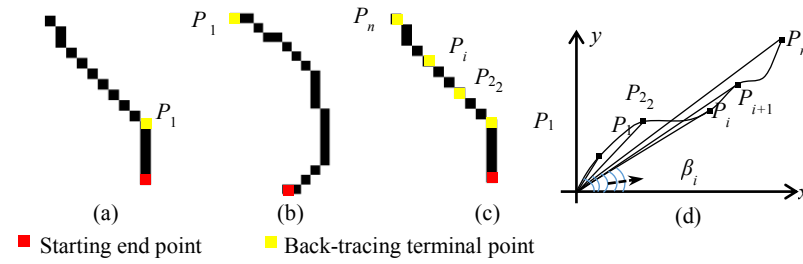
Based on the global topology properties(auto-enclosed, parallel and equidistant) and local properties(smoothing and continuous) of contour lines, in order to overcome the defects and faults of the other methods, the new proposed method designs and adopts a series of key algorithms, such as multiple incremental back-tracing to compute weighted average direction angle of end point, the max constraint angle control mechanism based on multiple gradient ranks, combination of weighted Euclidean distance and deviation angle between end points to determine the optimum matching point, bidirectional parabola control, etc.

In order to enforce the global and local restraint for end points' matching and reconnection, raise the accuracy and completion rate, and enhance the ability of dealing with all kinds of complicated samples of contour line image, the proposed method emphasizes its stability and adaptability throughout its procedure.

#### 3.1 Multiple incremental back-tracing

Previous methods usually detect and compute end points' directional angle by back-traceing fixed number of pixels from each end point, and use the linked-line between each end point and its back-tracing terminal point to construct directional angle. Back-tracing fixed number of pixels is inadequate and may cause some problems. For example, the direction of end point seems to be on  $x$ -axis or  $y$ -axis. It is very difficult to match with its pair which may not be on the  $x$ -axis or  $y$ -ordinate due to the property that contour lines generally have opposite and parabolic directions. In addition, the amount of back-traceing pixels is difficult to fix. If too little pixels are used for back-tracing, the main tendency of a contour line may be ignored (Figure 3(a)). On the contrary, if too many pixels are used, the information of local direction of a contour line would be difficult to be reflected (Figure 3(b)). Therefore, only one-time back-tracing fixed amount of pixels tends to cause wrong results and has its uncertainty. Because this, we put forward a algorithm of multiple incremental back-tracing and weighted mean value to get end points' directional an-

gle(Figure 3 (c) (d)), aiming at applying multiple incremental back-tracing to obtain weighted mean value as end points' directional angle, so that the results' stability and accuracy are further guaranteed.



**Fig .3.** (a) go back with too little pixels; (b) go back with too much pixels; (c) multiple incremental back-tracing; (d) the algorithm of multiple incremental back-tracing.

In figure3(d),  $P_1$  means 1st back-tracing by  $l_1$  pixels,  $P_2$  means 2nd back-tracing by  $l_2$  pixels,  $P_i$  means  $i$ -th back-tracing by  $l_i$  pixels,  $P_{i+1}$  means  $(i+1)$ -th back-tracing by  $l_{i+1}$  pixels,  $P_n$  means  $n$ -th back-tracing by  $l_n$  pixels. So this algorithm's main idea is: firstly build a back-tracing pixels dataset  $L\{l_1, l_2, \dots, l_i, \dots, l_n\}$ , which stores amount of increasing back-tracing pixels, place restrictions on minimum number  $\min\{l_i\}$  and maximum number  $\max\{l_i\}$  based on each contour line's actual length which end point  $P$  belongs to; next execute back-tracing to locate a terminal end point  $P_i$ , compute linked-line's directional angle between  $P$  and  $P_i$ , lastly compute weighted mean value of directional angles' data set  $\beta\{\beta_1, \beta_2, \dots, \beta_n\}$ , then the result would be assigned to ultimate directional angle value.

The formulas for computing ultimate directional angle may be expressed as shown below (1)(2) (3):

$$\bar{\beta} = \frac{\sum_{i=1}^n \beta_i \cdot f_i}{\sum_{i=1}^n f_i} \quad (1)$$

$$\beta_i = \text{atan2}((y_i - y_0)/(x_i - x_0)) \quad (2)$$

$$f_i = \frac{l_i}{\sum_{i=1}^n l_i} \quad (3)$$

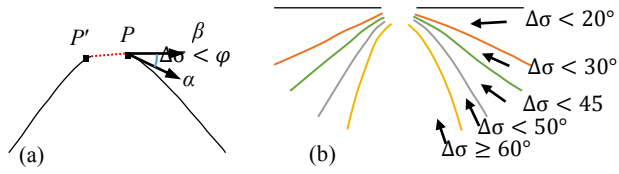
Let  $f_i$  represent distance's weight of  $PP_i$ ,  $\beta_i$  represent directional angle  $i$ th's back-training,  $(x_0, y_0)$  represent starting end point  $P$ ,  $(x_i, y_i)$  represent  $i$ th's back-tracing terminal pixel  $P_i$ .

Back-tracing pixels' dataset  $L\{l_1, l_2, \dots, l_i, \dots, l_n\}$  should be built flexibly according to broken contour line's actually length. For example, set  $L = \{3, 5, 7, 9, 11, 13, 15\}$ , thus seven times' back-tracing would be executed. Besides, there are some other conditions should be noticed:  $\min\{l_i\} = 3$ ,  $\max\{l_i\} = 15$ , with  $\max\{\phi_i\} < ll$ ,  $ll$  represents broken contour line's actual length.

This algorithm takes both broken contour line's local geometrical properties and global tendency into account. Therefore, its computational results possess higher accuracy, eliminating randomness and uncertainty caused by single back-tracing.

### 3.2 Max constraint angle control mechanism

As is shown in Figure 4(a), Let  $\alpha$  is the directional angle of end point  $P(x_0, y_0)$ , which indicates the general progressive tendency of the broken contour line from this starting point, let  $\beta$  is the inclined angle of connecting line between  $P$  and  $P'(x_i, y_i)$ , which indicates the progressive tendency of  $PP'$  similarly. Thus the deviation angle  $\Delta\sigma$ , with  $\Delta\sigma = |\beta - \alpha|$ , could directly reflect the transitional smoothing degree while two matched end points connecting with each other. According to the geometrical properties of contour line, matching and connecting between two end points ought to be gently and smoothing, the acute transitional cases, such as closed angle and acute angle, are disallowed (Figure 4(b) cases with  $\Delta\sigma \geq 60^\circ$ ). Therefore, it is very useful to construct max constraint angle condition to control the deviation angle, let  $\phi$  represent max constraint angle, which can help to control local transitional smoothing degree, and avoid abnormal transitional cases effectively, like closed angle, acute angle and so forth.



**Fig. 4.** (a) Max constraint angle's computation; (2) End points' connecting and transitional cases

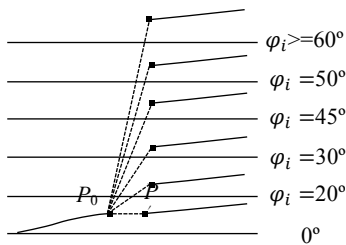
The formula for computing deviation angle and max constraint angle condition may be expressed as shown below (4):

$$\Delta\sigma = |\beta - \alpha| < \varphi \quad (4)$$

The condition of max constraint angle focuses too much on single end point's local constraint, lacking of global controlling function based on whole map. Maybe a given max constraint angle could solve part of broken contour lines' transition when connecting them, but the caps and contour line's shapes in the whole map are various and complicated, then setting single max constraint angle is deficient and not comprehensive enough. If the value is much bigger, this means that the constraint is too loose and liable to lead to intersecting errors. On the contrary, much smaller value would exclude a lot of end points which seem to be against to the condition but is fit for matching and connecting. So it is necessary to build a flexible mechanism to strength applicability, for this purpose, we propose max constraint angle control mechanism based on the gradient ranks.

This mechanism's main idea is: firstly, build a max constraint angle dataset  $T = \{\varphi_0, \varphi_1, \varphi_2, \dots, \varphi_n\}$ , with  $\varphi_i (i = 0, 1, 2, \dots, n)$  is an incremental element in  $T$ ; then use  $\varphi_i$  as a fixed value to control transitional smoothing degree between two end points; next, move  $\varphi_i$  to next element to control unsolved end points until the last element  $\varphi_n$  has been used.

On one side, the dataset  $T$  should be flexible and comprehensive based on considering all complicated cases of the whole binary image, each element's value is ascending step by step, and restrain the smoothing degree from small to large. On the other side, each element  $\varphi_i$  implies the local transitional smoothing degree under a fixed range. As we see in Figure 5,  $T$  could be constructed according to whole map's morphological characteristics, for example, set  $T = \{0^\circ, 20^\circ, 30^\circ, 45^\circ, 50^\circ, 60^\circ\}$ , then Figure 5 shows a variety of transitional morphology under different deviation angles. When  $\varphi_i \leq 45^\circ$ , the transition between two match end points is suitable and conforms to common cases; when  $45^\circ < \varphi_i \leq 60^\circ$ , the transition is unnatural and liable to cause errors, decision should be cautious when using this scope; when  $\varphi_i > 60^\circ$ , the transition is extremely steep, that is to say  $\max\{\varphi_i\}$  should be smaller than  $60^\circ$ .



**Fig. 5.** Different smoothing degree under gradient ranks' max constraint angle

In fact, this mechanism firstly focuses on the most gently and simple broken contour lines, and intends to accomplish those end points' matching; thereafter, it increases max constraint angle value to solve the remaining end points gradually. So this mechanism realizes a procedure from simple to complex, from gently to steep, reveals local and global geometrical properties of contour lines, could eliminate intersects and other errors.

### 3.3 Combination of weighted Euclidean distance and angle to compute matching measurement

Previous processing algorithms have guarantee smooth transition when connecting matched end points, but there is still not a ultimate measurement indicator for completing right matching between either end points and reserved end points. In other words, it is of crucial importance to design an algorithm to provide ultimate measurement indicator. As we all know, distance and deviation angle are the two primary impact factors, whereas Samet's method (Samet and Hancer 2012) use the minimum  $y$ -ordinate distance as ultimate measurement indicator to decide optimal match end point. That algorithm is simple and computational-cost is very low, though, the results are unreliable. For example, the minimum  $y$ -ordinate distance is not equal to minimum spatial distance, conversely the minimum  $x$ -ordinate distance may be very large, so Samet's method is likely to cause wrong connections (See in Figure 2 red drawing). Euclidean distance is the actual plane distance, and is superior to  $y$ -ordinate distance when judging near and far, at the same time, computing accuracy is improved dramatically. In this paper, the  $y$ -ordinate distance is replaced by Euclidean distance. On this basis, we propose an algorithm which could gain a quantitative comparison measurement to determine optimal match end point based on weighted combination of Euclidean distance and deviation angle.

This algorithm set weightings on Euclidean distance and deviation angle partly, and combine them to compute a reference value, which is set as a quantitative comparison measurement, and the minimum value gained from all values assigns the optimal matching end point. Distance's weighting and deviation angle's weighting reflect its importance and influencing factor in the computing of matching measurement. The more approaching to each other, the more possible it is to match successfully; the smaller the deviation angle is, the more smoothing the transition of connection is. All in all, the minimum matching measurement has its reasonability and validity.

The main idea of this algorithm is: computing the Euclidean distance  $ds_i$  and deviation angle  $\Delta\sigma_i$  between the center end point  $P_0$  and the chosen end points  $P_i (i = 0, 1, \dots, k)$  firstly; next computing the value  $D_i$  by combining weighted distance and deviation angle; at last, detect the matched end points by choosing  $D_{min} = \min\{D_i\} (i = 0, 1, \dots, k)$ . Finally,  $D_i$  may be discretized with formula (5):

$$D_i = \Delta\sigma_i * w_1 + ds_i * w_2 \quad (i = 0, 1, \dots, k) \quad (5)$$

Let  $w_1$  represents deviation angle's weighting, represents Euclidean distance of  $P_0P_i$ ,  $w_2$  represents distance's weighting.

The measuring units of distance and angle are different, and directly combining operation is disallowed. It is essential to convert angle to distance to make them to be under the same measuring unit, so we finish this conversion. The main idea of the conversion is: (1) convert deviation angle to central angle of a circle under a given radius; (2) compute corresponding arc length; (3) replace the deviation angle with arc length to compute matching measurement. The converting formulas are below (6)(7):

$$L_i = |\Delta\sigma_i| * r \quad (i = 0, 1, \dots, k) \quad (6)$$

$$D_i = L_i * w_1 + ds_i * w_2 \quad (7)$$

Let  $r$  represents a given circle radius,  $\Delta\sigma_i$  represents converted central angle,  $L_i$  represents arc length.

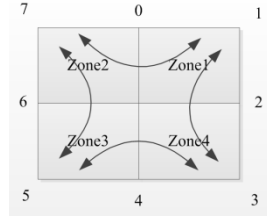
Generally, the weightings  $w_1$  and  $w_2$  assigned to distance and angle should flexible and variable. Gradient ranks' max constraint angle mechanism controls transitional smoothing degree effectively and excludes a lot of end points with acute angle, then more attention should be paid to spatial distance near or far, in case of long distance cause intersection and other errors. Therefore, distance's weighting should be emphasized and superior to angle's weighting, the latter is a beneficial supplement to the former. For example, set  $w_1 = 0.2$ ,  $w_2 = 0.8$ . These two values should be varied according to the matching results.

### 3.4 Bidirectional control based on parabola

Although previous algorithm and mechanism focus too much on right matching and smoothing transition from global to local, they are insufficient to deal with end points presenting acute angle parabola shape (Fig1(b) marked by black rectangle). After color map's scanning and segmenting, there are a lot of end points presenting acute angle parabola shape in binary image. In Samet's method, clockwise end points' direction codes matching algorithm was

designed and applied to solve this problem. But only part of these end points have been solved, a lot of end points at counterclockwise direction can not be solved and are liable to lead to errors.

Bidirectional codes matching algorithm based on parabola shape is put forward in this article. Its main idea is to extent clockwise and counterclockwise bidirectional codes matching (see Fig6 and Table 1), in order to improve the accuracy and comprehensiveness in direction codes matching. This algorithm is an important supplement to solve complicated and acute parabola end points, raising completeness rate of whole image effectively.



**Fig 6.** Bidirectional matching of end points based on parabola shape

**Table 1.** End points' directional code matching based on bidirectional parabolic shape

End point direction's code	Selectable end point's direction code
1	3 or 7
3	5 or 1
5	7 or 3
7	1 or 5

### 3.5 The executing procedure of proposed method

The proposed method's general idea is that all end points are marked and fetched firstly, multiple back-tracing to compute one end point's directional angle, use gradient ranks of max constrained angle to control smoothing degree of reconnection, and compute minimum matching measurement value to determine optimal matching end point. At last, Newton interpolation algorithm is used to reconnect broken contours. The detailed executing procedure flows are listed as follows:

Step1: Preprocess the color scanned map: segment and threshold color image, filter noisy pixels, thin and prune the binary image.

Step2: Set threshold, initialize all parameters, including: maximum search window, extension step, weight factors of distance and directional angle.

Step3: Search all end points over the whole binary image, form end point array  $\{P_0, \dots, P_i, \dots, P_n\}$ , set initial max constraint angle  $\varphi_i$ ,  $i = 0$ .

Step4: Start the procedure of end points matching for  $\{P_0, \dots, P_i, \dots, P_n\}$  from end point  $P_0$  based on max constraint angle  $\varphi_i$ :

- ① Detect a square mask  $h \times h$  and assign  $P_0$  as the center of the square mask, go back-tracing to detect another end point which is on the other end of the same small contour line. If any end point comes across, do not use this end point, mark and exclude it.
- ② Detect other end points within the  $h \times h$  square mask, form a chosen end point array  $\{P_1, \dots, P_j, \dots, P_m\}$ .
- ③ Compute  $P_0$ 's directional angle  $\alpha$ , the temporal connecting line's inclined angle  $\beta$  constructed by  $P_0$  and its chosen end point  $P_j$ ,  $P_j \in \{P_1, \dots, P_j, \dots, P_m\}$ , the deviation angle  $\Delta\sigma_j$  ( $j = 1, 2, 3, \dots, m$ ) between  $\alpha$  and  $\beta$ , then judge whether  $\Delta\sigma_j < \varphi_i$ , if yes remain it, else remove it from  $\{P_1, \dots, P_j, \dots, P_m\}$ , judge whether  $\{P_1, \dots, P_j, \dots, P_m\}$  have been traversed, if yes form a new chosen end point array  $\{P_1, \dots, P_j, \dots, P_{m'}\}$ , go to Step ④.



- ④ Compute the Euclidean distance  $ds_j$  between  $P_0$  and its chosen end point  $P_j$ ,  $P_j \in \{P_1, \dots, P_j, \dots, P_m\}$ , combine weighted  $ds_j$  and  $\Delta\sigma_j$  to calculate  $P_0$ 's matching measurement  $D_j$ .
  - ⑤ Judge whether  $D_j$  reaches to minimum. When  $D_i$  reaches to minimum  $D_{min}$ , its corresponding end point  $P'$  is regarded as the object point.
  - ⑥ reconnect these two matched end points  $P_0, P'$  by utilizing Newton Interpolation algorithm, and remove them from  $\{P_0, \dots, P_i, \dots, P_n\}$ , refresh the array.
  - ⑦ Judge whether it remains un-filling caps in the binary image, if yes go to Step5, else go to Step 7.
- Step5: Judge whether all max restrained degree ranks have been traversed, if yes go to Step 6, else set  $\varphi_k = \varphi_{k+1}$ , go to Step 4.

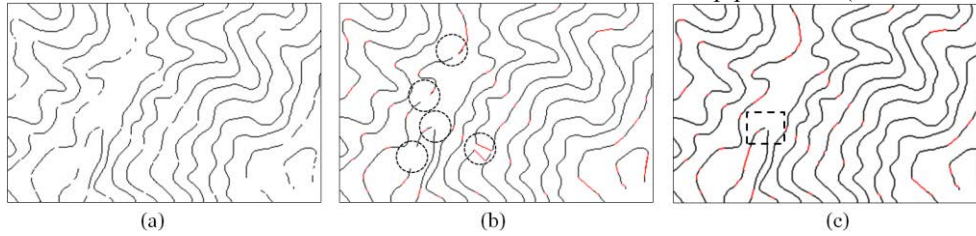
Step6: Use bidirectional parabola shape end points' direction code matching algorithm to reconnect end points which possess the feature of acute angle parabola.

Step7: Results are summarized and programme exit.

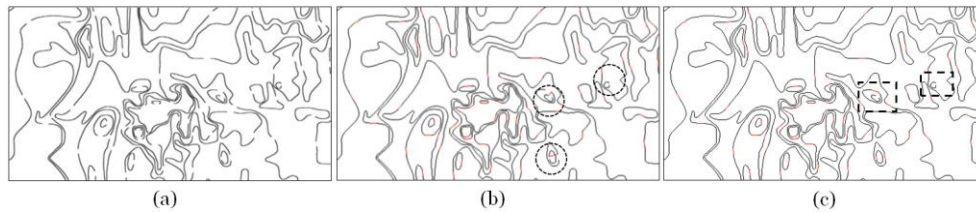
## 4 Experiments and analysis

In order to test the effectiveness, stability and applicability, the proposed method has been applied to many different samples for analyzing and comparing with other methods. Due to the paper's length limit, three images of relatively complex topographic maps in different quality are used in this paper, and the experimental results' evaluations are presented from two aspects: one is qualitative evaluation, the other is the analysis of measurement indicators, including the number of finished connections, number of false connections, completeness rate, accuracy rate, etc.

We have used Samet's method and proposed method to process three typical images respectively. Although Fig7(b), Fig8(b) and Fig9(b) demonstrate that Samet's method could implement reconnecting most of broken contour lines, obvious errors (marked with black circle), for example intersects and failing to fill up bigger or steep caps, are still unresolved. On the contrary, Fig7(c), Fig8(c) and Fig9(c) indicate that the proposed method has almost reconnected all the broken contour lines, especially realized matching almost all end points with each other correctly, and the little remained unconnected broken contour lines hold steep parabolas (marked with black rectangle).

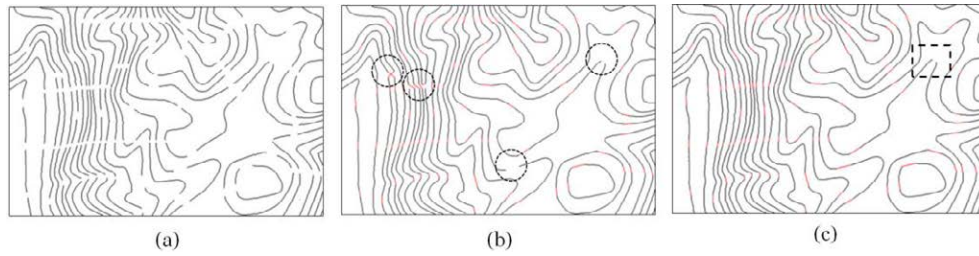


**Fig. 7.** Sample 1: (a) pre-processed binary image (Source: Ghircoias and Brad 2011); (b) result of Samet's method [10]; (c) result of proposed method



**Fig. 8.** Sample 2: (a) pre-processed binary map (Source: Pouderoux and Spinello 2011); (b) result of Samet's method; (c) result of proposed method





**Fig. 9.** Sample3: (a) pre-processed binary map; (b) result of Samet's method; (c) result of proposed method

In order to analyze and compare with other method, the following numerical indicators are used: (1) total gaps; (2) the number of filled gaps; (3) the number of unresolved gaps; (4) the number of wrong connections; (5) completeness rate; and (6) accuracy rate. Here let  $T$  represents the number of total gaps,  $T_1$  represents the number of filled gaps,  $T_2$  represents the number of not filled gaps,  $C_1$  represents the percent of reconnected number,  $C_2$  represents accuracy percent. In general,  $T$  is very difficult to be obtained from contour lines' images automatically, which is acquired manually;  $T_1$  could be recorded in processing automatically; and  $T_2$  merely could be acquired by checking the original image and resulting image manually. The formulas of computing  $C_1$  and  $C_2$  are given below (8)(9):

$$C_1 = \frac{|T_1 \cap T|}{|T|} \quad (8)$$

$$C_2 = \frac{|T_1 \cap T_2|}{|T_1|} \quad (9)$$

Analysis and comparison results of proposed method and Samet's method are given in Table 2, and all the numerical indicators come from previous three experimental samples(Fig 8, Fig 9 and Fig 10).

**Table 2** Analysis and comparison results of proposed approach and Samet's method

Map name	Map resolution	Total gaps	Method	$T_1$	$T_1 - T_2$	$C_1(\%)$	$C_2(\%)$
Fig 7(a)	581*431	53	Samet's method	50	4	94	96
			proposed method	52	0	98	100
Fig 8(a)	1278*903	70	Samet's method	66	3	90	95
			proposed method	68	0	97	100
Fig 9(a)	704*468	101	Samet's method	91	6	84	93
			proposed method	100	0	99	100

## 5 conclusion

In this paper, a novel method is proposed for reconstruction of broken contour lines. Because all existed techniques and methods can still not get satisfactory results yet, the proposed method is dedicated to achieve good stability and effectiveness in processing different quality images. This method designs and utilizes a series of key algorithms and mechanisms based on the geometrical and topological properties of contour lines.

The excellent contribution of the proposed method is that it can eliminate the errors of other methods, increase the accuracy and completeness rate in end points' matching and reconnecting effectively. For the difficult gaps which hold acute angle parabola, proposed method utilizes bidirectional parabola control algorithm to solve this problem. At last, comparative experiments between proposed method and the other representative Samet's method have been accomplished, the results indicate that proposed method could get higher rate of accuracy and completeness, possess the merits of better stability, adaptability and applicability. Next work in progress of this paper is to decrease computational complexity and increase efficiency.

## References

- P. Arrighi, P. Soille(1999). From scanned topographic maps to digital elevation models, in: Proceedings of Geovision'99.
- A. Khotanzad, E. Zink(2003). Contour line and geographic feature extraction from USGS color topographical paper maps, IEEE Trans. Pattern Anal. Mach. Intell. 25 (1), 18–31.
- L. Eikvil, K. Aas, H. Koren(1995). Tools for interactive map conversion and vectorization[J], International Conference on Document Analysis and Recognition, Third International Conference on Document Analysis and Recognition, 2,927–930
- L. Mun San, S. Mat Yatim, N. Azam MD Sheriff, N. Isrozaidi bin Nik Ismail(2004). Extracting contour lines from scanned topographic maps, in: International Conference on CGIV'2004.
- N. Isrozaidi Nik Ismail, N. Azam MD Sheiriff(2006). A Novel Technique for Contour Reconstruction To Dem. PhD thesis, Universiti Teknologi Malaysia.
- D. Xin, X. Zhou, H. Zheng(2006). Contour line extraction from paper-based topographic maps, J. Inform. Comput. Sci. 1 (5), 275–283.
- D. Xin, X. Zhou, Y. Shi(2006). Extraction algorithm of contour lines based on generalized gradient vector flow snakes model[J]. Journal of Computer-aided Design&Computer Graphics, 18(7), 982–986.
- N. Amenta, M. Bern, D. Eppstein (1998). The crust and the beta-skeleton: combinatorial curve reconstruction[J]. Graphical Models Image Process. 60 (2), 125–135.
- T. Ghircoias, R. Brad(2011). Contour lines extraction and reconstruction from topographic maps[J], Ubiquitous Comput. Comm. J. 6 (2), 681–691.
- J. Pouderoux, S. Spinello(2007). Global contour lines reconstruction in topographic maps, Proc. of the Ninth International Conference on Document Analysis and Recognition, (ICDAR 2007) 2, 779–783.
- R. Samet, E. Hancer (2012). A new approach to the reconstruction of contour lines extracted from topographic maps[J]. J. Vis. Commun. Image R. 23, 642–647.
- Xue-Qian Huang, Peng Hu, Yi-Duo Bai( 2006). An improved method for connecting broken contour [J]. Science of Surveying and Mapping, 31(1), 111–113.
- Mohan P Pradhan, M K Ghose, Pooja S Rai, Nilanjan Mukherjee( 2013). Knowledge based contour line reconnection techniques[J]. International Journal of Computer Applications ,65(9),37–41
- D.P.J. Giesen(2000). Curve Reconstruction. Phd thesis. Swiss Federal Institute of Technology.
- J.R. Parker(1997). Algorithms for Image Processing and Computer Vision, Wiley Computer Publishing, New York.